# Real-time Kalman filter implementation for active feedforward control of time-varying broadband noise and vibrations

**S. van Ophem** [1] and **A.P. Berkhoff** [1,2]
[1] University of Twente, Faculty EEMCS, PO Box 217, 7500AE Enschede, The Netherlands
e-mail: **a.p.berkhoff@utwente.nl**

[2] TNO Technical Sciences, Acoustics and Sonar, PO Box 96864, 2509JG The Hague, The Netherlands

## Abstract
Tracking behavior and the rate of convergence are critical properties in active noise control applications with time-varying disturbance spectra. As compared to the standard filtered-reference Least Mean Square (LMS) algorithm, improved convergence can be obtained with schemes based on preconditioning, affine projections, Recursive Least Squares (RLS), and other methods. The RLS method potentially leads to very fast convergence but straightforward implementations may suffer from round-off errors and suboptimal tracking behavior. In this paper a Kalman filter approach is used which, as compared to RLS, includes a covariance weighting for model output errors in order to improve tracking behavior and robustness. An orthogonal transformation scheme was used to reduce the influence for round-off errors of the Kalman recursions. In this paper an extension is given for multiple input multiple output systems, and results of a real-time implementation are presented. Conclusions are given regarding the suitability for real-time implementation of previous formulations. Different parameterizations of the secondary path model between the active control source and the error sensor are compared. Real-time results demonstrate the rapid convergence for reduction of noise in a duct, as well as numerical stability during extended operation intervals.

## 1 Introduction

Filtered-reference and filtered-error least means squares algorithms based on approximate, instantaneous gradients are widely used for adapting an Active Noise Control (ANC) system. The algorithms are relatively simple and robust, but one of the biggest drawbacks of the algorithms are the low rate of convergence leading to slow adaption to changes in the primary path. The assumption that is used is that the filter coefficients are changing slowly in comparison to the timescale of the plant dynamics, see Elliott [1]. Several approaches have been suggested to improve the speed of convergence of least-mean-square based algorithms, such as the modified fx-LMS algorithm proposed by Bjarnason [2], Fast Affine Projections, Preconditioned LMS [1] and other methods.

Recursive Least Squares (RLS) algorithms have a faster rate of convergence, but require more computational effort. A modified RLS algorithm has been derived by Flockton [3], which has a similar structure as a modified LMS-algorithm. The disadvantages of this algorithm are the initial overshoot when the filter is turned on and slow tracking behavior. Sayed *et al.* [4] have shown that the RLS filter is a special case of a Kalman filter. A SISO Kalman filter was described by Fraanje [5] in an ANC context, in which it was shown that there is no initial overshoot before convergence when a properly tuned Kalman filter is used for an ANC application. This filter estimates the state of the secondary path and the filter coefficients and takes

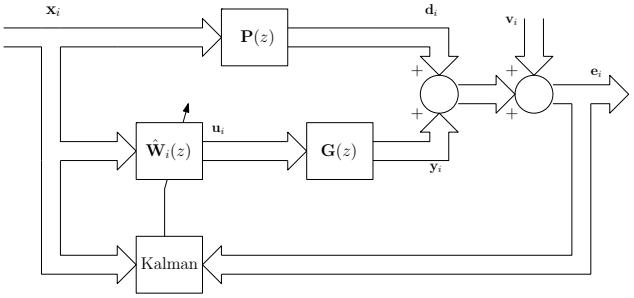


Figure 1: Block diagram of a MIMO ANC system with a Kalman filter.

uncertainty of the state and uncertainty in measurements into account, which explains the abscence of the overshoot in the convergence curve. As compared to the RLS filter, tracking behavior is potentially improved because system uncertainties are taken into account in the algorithm.

This paper presents results of a multiple input multiple output Kalman filter as derived for an ANC system. An extension of the algorithm is made with a state space description of the secondary path in output normal form. This requires less computational effort and is numerically more robust. Simulations will be shown of a multiple input multiple output implementation of the Kalman recursions in free field conditions. The performance of the algorithm was tested in a real-time experiment in which the goal was to minimize the noise at the end of a duct for time-varying signals.

# 2 MIMO Kalman filter

## 2.1 Model description

Consider an ANC system which has $N_x$ reference channels, $N_u$ control channels and $N_e$ error channels with the reference signal vector $\mathbf{x}(i) \in \mathbb{R}^{N_x}$, the control signal vector $\mathbf{u}(i) \in \mathbb{R}^{N_u}$, and the error signal vector $\mathbf{e}(i) \in \mathbb{R}^{N_e}$, in which $i$ is the time instance. In this paper, the time index $i$ is indicated either as a subscript or between parentheses. It is assumed that there is no feedback from the actuators to the reference microphones, so the control system can be seen as a purely feed-forward system, as shown in Fig. 1. In this figure $\mathbf{P}(z)$ represents the primary path from the reference microphone to the error microphone, $\mathbf{G}(z)$ represents the secondary path from the secondary actuator to the error microphone and $\hat{\mathbf{W}}_{i,\mathrm{FIR}}(z)$ is the controller, adapted by a Kalman filter.

The adaptive controller has a feed-forward structure and is described by the matrix $\hat{\mathbf{W}}_{i,\mathrm{FIR}}(z) \in \mathbb{R}^{N_u \times N_x}$ consisting of FIR-filters with $n_w$ filter coefficients. The $(k,l)_{th}$ term of this matrix, with $0 \leq k \leq N_u,\ 0 \leq l \leq N_x$ can be described by:

$$\hat{\mathbf{W}}^{(k,l)}_{i,\mathrm{FIR}}(z) = \hat{w}^{(k,l)}_0(i) + \hat{w}^{(k,l)}_1(i)z^{-1} + ...\hat{w}^{(k,l)}_{n_w-1}(i)z^{-n_w-1}. \tag{1}$$

The individual filter coefficients can be organized as follows:

$$\hat{\mathbf{w}}^{(k,l)}(i) = \left[\ \hat{w}^{(k,l)}_0(i) \quad ... \quad \hat{w}^{(k,l)}_{n_w-1}(i)\ \right]^T \in \mathbb{R}^{n_w \times 1}, \tag{2}$$

$$\hat{\mathbf{w}}^{(k)}(i) = \left[\ \hat{\mathbf{w}}^{(k,1)}(i)^T \quad ... \quad \hat{\mathbf{w}}^{(k,N_x)}(i)^T\ \right]^T \in \mathbb{R}^{n_w N_x \times 1}, \tag{3}$$

$$\hat{\mathbf{W}}(i) = \left[\ \hat{\mathbf{w}}^{(1)}(i) \quad ... \quad \hat{\mathbf{w}}^{(N_u)}(i)\ \right] \in \mathbb{R}^{n_w N_x \times N_u}. \tag{4}$$

$$\hat{\mathbf{w}}^{(k)}(i) = \left[ \begin{array}{ccc} \hat{\mathbf{w}}^{(k,1)}(i)^T & \ldots & \hat{\mathbf{w}}^{(k,N_x)}(i)^T \end{array} \right]^T \in \mathbb{R}^{n_w N_x \times 1}, \tag{5}$$

The $N_x$ vectors with the last $n_w$ steps of the $n_x$-th reference signal $\mathbf{x}_{n_w}^{n_x}(i)$ are stacked in the vector $\mathbf{x}_{n_w}(i)$. The resulting control signals are (using $\mathrm{vec}(\mathbf{ABC}) = (\mathbf{C}^T \otimes \mathbf{A})\mathrm{vec}(\mathbf{B})$ [6]):

$$\begin{aligned} \mathbf{u}(i) &= -\hat{\mathbf{W}}^T(i)\mathbf{x}_{n_w}(i) = -\mathrm{vec}(\mathbf{x}_{n_w}^T(i)\hat{\mathbf{W}}(i)I_{N_u}) \\ &= -(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i))\hat{\mathbf{w}}(i). \end{aligned} \tag{6}$$

In this equation $\boldsymbol{I}_{N_u}$ is the identity matrix of size $N_u$. When the controller reaches its optimal value $\mathbf{w}^o(i)$, the control signal vector is:

$$\mathbf{u}^o(i) = -(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i))\mathbf{w}^o(i). \tag{7}$$

## 2.2 Augmented state space description

The MIMO ANC problem is written in a state space form, where the purpose of the FIR controller is to minimize the error $\mathbf{e}(i) = \mathbf{d}(i) + \mathbf{y}(i) + \mathbf{v}(i)$, in which $\mathbf{d}(i)$ is the influence of the primary paths on the error microphones, $\mathbf{y}(i)$ is the influence of the secondary path on the error sensors and $\mathbf{v}_i$ is assumed to be a Gaussian white noise signal vector, corrupting the measurement of the error sensors. The error is minimized when the FIR filters are adjusted to their optimal values $\mathbf{W}_i^o(z)$, so that $\mathbf{d}(i) = -\mathbf{y}(i)$. As stated by Sayyarrodsari [7] the purpose of the active noise control algorithm is to make a model of the primary path with the series connection of the FIR filter matrix and the secondary path. Therefore the primary path can be approximated by a series connection of the optimal filter $-\mathbf{W}_i^o(z)$ and the secondary path $\mathbf{G}(z)$, as shown in Fig. 2. A noise vector $\mathbf{n}(i)$ is included to account for modeling uncertainties. Using the methods of [8] and defining the augmented state vector

$$\boldsymbol{\chi}(i) = \left[ \begin{array}{c} \mathbf{w}(i) \\ \boldsymbol{\theta}(i) \end{array} \right], \tag{8}$$

the augmented state space description can be written as:

$$\begin{aligned} \left[ \begin{array}{c} \mathbf{w}(i+1) \\ \boldsymbol{\theta}(i+1) \end{array} \right] &= \boldsymbol{A}(i) \left[ \begin{array}{c} \mathbf{w}(i) \\ \boldsymbol{\theta}(i) \end{array} \right] - \boldsymbol{B}(i)\hat{\mathbf{w}}(i) + \boldsymbol{H}(i)\mathbf{n}(i)\ , \\ \mathbf{e}(i) &= \boldsymbol{C}(i) \left[ \begin{array}{c} \mathbf{w}(i) \\ \boldsymbol{\theta}(i) \end{array} \right] - \boldsymbol{D}(i)\hat{\mathbf{w}}(i) + \mathbf{v}(i), \ \left[ \begin{array}{c} \mathbf{w}(0) \\ \boldsymbol{\theta}(0) \end{array} \right] = \left[ \begin{array}{c} \mathbf{w}^o(0) \\ \boldsymbol{\theta}(0) \end{array} \right] \end{aligned} \tag{9}$$

Introducing a forgetting factor $0 \ll \lambda \leq 1$ [8], the full state space description for a MIMO ANC feed-forward system can be written as:

$$\mathbf{A}(i) = \left[ \begin{array}{cc} \lambda^{-1/2}\boldsymbol{I}_{n_w N_u N_x} & 0_{n_w N_u N_x \times n_s} \\ \mathbf{B}_s(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i)) & \mathbf{A}_s \end{array} \right], \tag{10}$$

$$\mathbf{B}(i) = \left[ \begin{array}{c} 0_{n_w N_u N_x \times n_w N_u N_x} \\ -\mathbf{B}_s(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i)) \end{array} \right], \tag{11}$$

$$\mathbf{C}(i) = \left[ \begin{array}{cc} \mathbf{D}_s(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i)) & \mathbf{C}_s \end{array} \right], \tag{12}$$

$$\mathbf{D}(i) = -\mathbf{D}_s(\boldsymbol{I}_{N_u} \otimes \mathbf{x}_{n_w}^T(i)), \tag{13}$$

$$\mathbf{H} = \left[ \begin{array}{c} 0_{n_w N_u N_x \times N_e N_x} \\ \mathbf{H}_s \end{array} \right], \tag{14}$$

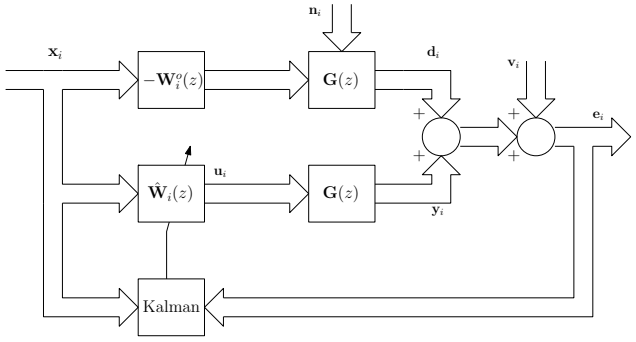


Figure 2: Block diagram of a MIMO ANC system with an approximate representation of the primary path.

leading to the state space description

$$\begin{aligned} \boldsymbol{\chi}(i+1) &= A(i)\boldsymbol{\chi}(i) + B(i)\hat{\mathbf{w}}(i) + H\mathbf{n}(i), \\ \mathbf{e}(i) &= C(i)\boldsymbol{\chi}(i) + D(i)\hat{\mathbf{w}}(i) + \mathbf{v}(i). \end{aligned} \tag{15}$$

## 2.3 Kalman filtering

To estimate the state of the ANC system a Kalman filter is used, in which the Kalman filter explicitly takes the covariances from the noise vectors $\mathbf{n}_i$ and $\mathbf{v}_i$ into account and gives a minimum variance estimate of the state. The initial state is assumed to be uncorrelated with the noise terms $\mathbf{n}_i$ and $\mathbf{v}_i$. Also the inididual noise terms are assumed to be uncorrelated, resulting in

$$\mathrm{E}\left( \begin{bmatrix} \boldsymbol{\chi}(0) \\ \mathbf{n}(0) \\ \mathbf{v}(0) \end{bmatrix} \begin{bmatrix} \boldsymbol{\chi}(0) \\ \mathbf{n}(0) \\ \mathbf{v}(0) \end{bmatrix}^T \right) = \begin{bmatrix} \boldsymbol{\Pi}(0) & 0 & 0 \\ 0 & \mathbf{Q}\delta_{kl} & 0 \\ 0 & 0 & \mathbf{R}\delta_{kl} \end{bmatrix}. \tag{16}$$

In this equation $\delta_{ij}$ is the Dirac delta function and $\boldsymbol{\Pi}(0)$ is called the state covariance matrix, defined as

$$\boldsymbol{\Pi}(0) = \begin{bmatrix} \boldsymbol{\Pi}^{ww}(0) & \boldsymbol{\Pi}^{w\theta}(0) \\ \boldsymbol{\Pi}^{\theta w}(0) & \boldsymbol{\Pi}^{\theta\theta}(0) \end{bmatrix}, \tag{17}$$

with $\boldsymbol{\Pi}_0^{ww} = \mathrm{E}[\mathbf{w}(0)\mathbf{w}^T(0)]$, $\boldsymbol{\Pi}_0^{w\theta} = \boldsymbol{\Pi}_0^{\theta w^T} = \mathrm{E}[\mathbf{w}(0)\boldsymbol{\theta}^T(0)]$, $\Pi_0^{\theta\theta} = \mathrm{E}[\boldsymbol{\theta}(0)\boldsymbol{\theta}^{\mathrm{T}}(0)]$.
Furthermore $\mathbf{Q}$ and $\mathbf{R}$ are the noise covariance matrices, and are given by

$$\mathbf{Q} = \begin{bmatrix} Q_{11} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & Q_{N_x N_e} \end{bmatrix}, \tag{18}$$

$$\mathbf{R} = \begin{bmatrix} R_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & R_{N_e} \end{bmatrix}. \tag{19}$$

The estimate of the control coefficients $\hat{\mathbf{w}}(i)$ and the estimate of the state vector $\hat{\boldsymbol{\theta}}(i)$ are combined in the estimate of the augmented state vector

$$\hat{\boldsymbol{\chi}}(i) = \begin{bmatrix} \hat{\mathbf{w}}(i) \\ \hat{\boldsymbol{\theta}}(i) \end{bmatrix}, \tag{20}$$

Then the MIMO Kalman filter in covariance form is given by the following equations:

$$\begin{aligned}
\hat{\boldsymbol{\chi}}(0) &= 0_{n_w N_u N_x + n_s \times 1}, && (21)\\
\mathbf{P}(0) &= \boldsymbol{\Pi}(0), && (22)\\
\boldsymbol{\epsilon}(i) &= \mathbf{e}(i) - \mathbf{C}_s \hat{\boldsymbol{\chi}}(i) - \mathbf{D}_s \hat{\mathbf{w}}(i), && (23)\\
\mathbf{R}_e(i) &= \mathbf{R} + \mathbf{C}(i)\mathbf{P}(i)\mathbf{C}^T(i), && (24)\\
\mathbf{K}(i) &= \mathbf{A}(i)\mathbf{P}(i)\mathbf{C}^T(i), && (25)\\
\hat{\boldsymbol{\chi}}(i+1) &= \mathbf{A}_s \hat{\boldsymbol{\chi}}(i) + \mathbf{B}_s \hat{\mathbf{w}}(i) + \mathbf{K}(i)\mathbf{R}_e^{-1}(i)\boldsymbol{\epsilon}(i) && (26)\\
\mathbf{P}(i+1) &= \mathbf{A}(i)\mathbf{P}(i)\mathbf{A}^T(i) - \mathbf{K}(i)\mathbf{R}_e^{-1}(i)\mathbf{K}^T(i) + H\mathbf{Q}H^T. && (27)
\end{aligned}$$

For a derivation of these equations, see Sayed [9]. In these equations $\mathbf{P}(i)$ represents the covariance matrix of the state estimation error, $\boldsymbol{\epsilon}(i)$ the innovation vector, $\mathbf{R}_e(i)$ the error covariance matrix and $\mathbf{K}(i)$ the gain matrix. Since the state is augmented, some of these expressions can be reduced and partitioned as follows:

$$\begin{aligned}
\mathbf{P}(i) &= \begin{bmatrix} \mathbf{P}^{ww}(i) & \mathbf{P}^{w\theta}(i) \\ \mathbf{P}^{\theta w}(i) & \mathbf{P}^{\theta\theta}(i) \end{bmatrix}, && (28)\\
\mathbf{K}(i) &= \begin{bmatrix} \mathbf{K}^{w}(i) \\ \mathbf{K}^{\theta}(i) \end{bmatrix}, \\
\boldsymbol{\epsilon}(i) &= \mathbf{e}(i) - \mathbf{C}_s \hat{\boldsymbol{\theta}}(i), && (29)\\
\begin{bmatrix} \hat{\mathbf{w}}(i+1) \\ \hat{\boldsymbol{\theta}}(i+1) \end{bmatrix} &= \begin{bmatrix} \lambda^{-1/2}\hat{\mathbf{w}}(i) \\ \mathbf{A}_s \hat{\boldsymbol{\theta}}(i) \end{bmatrix} + \begin{bmatrix} \mathbf{K}^{w}(i) \\ \mathbf{K}^{\theta}(i) \end{bmatrix} \mathbf{R}_e^{-1}(i)\boldsymbol{\epsilon}(i), && (30)
\end{aligned}$$

in which $\mathbf{K}_i^w \in \mathbb{R}^{n_w N_u N_x \times N_e}$, and $\mathbf{K}_i^\theta \in \mathbb{R}^{n_s \times N_e}$. The straightforward implementation of Eqs. (21) - (27) gives a computationally demanding algorithm. A more efficient algorithm can be achieved by making use of the shift-invariance of the reference signals. Use of shift invariance properties, the resulting Fast Array descriptions, and a specific initialization leading to a reduced rank of the update scheme, are summarized in Ref. [8].

## 2.4 Shift invariance, Fast Array form and initialization

The shift invariance means that, for $1 \leq n \leq N_x$, the reference signal vector $\mathbf{x}_{n_w}^{(n)}(i+1)$ is just a shifted version of $\mathbf{x}_{n_w}^{(n)}(i)$ with one new term. For multiple reference signals this shift behavior can be expressed in the following equation:

$$\begin{aligned}
\mathbf{x}_{n_w}^T(i) &= \mathbf{x}_{n_w}^T(i+1)(I_{N_x} \otimes \mathbf{Z}_{n_w}) + \\
&\quad \begin{bmatrix} 0_{1\times n_w - 1} & x^{(1)}(i - n_w + 1) & \dots & 0_{1\times n_w - 1} & x^{(N_x)}(i - n_w + 1) \end{bmatrix}. && (31)
\end{aligned}$$

The matrix $\mathbf{Z}_{n_w} \in \mathbb{R}^{n_w \times n_w}$ is a first diagonal shift matrix and $\otimes$ is the Kronecker matrix product. Due to the shift invariance, the state matrices have to be augmented with zeros [8].

The relations between the augmented state space matrices on instance $i+1$ and $i$ can be used to derive a fast array Kalman filter. This implementation of the Kalman filter has the following advantages (Sayed [4]):

- Due to the shift-invariance the calculation efficiency is much better than the standard covariance Kalman filter.

- Numerical round-off errors are not amplified when transformations are applied, because the norms and angles of inner products are preserved, when transforming.
- The use of square root factors limits the dynamic range.

The reason for the high calculation efficiency of the fast array form of the Kalman filter is that the difference of the Riccati equation is updated instead of the Riccati equation itself:

$$d\tilde{\mathbf{P}}_i = \tilde{\mathbf{P}}_i - \mathbf{\Psi}\tilde{\mathbf{P}}_{i-1}\mathbf{\Psi}^T. \tag{32}$$

When proper initial conditions for the covariance matrix are chosen, $d\tilde{\mathbf{P}}_i$ can be of very low rank, see Ref. [9]. Also the following difference equations are defined :

$$\begin{aligned} d\tilde{\mathbf{R}}_{e,i} &= \tilde{\mathbf{R}}_{e,i} - \tilde{\mathbf{R}}_{e,i-1}, & (33)\\ d\tilde{\mathbf{K}}_i &= \tilde{\mathbf{K}}_i - \mathbf{\Psi}\tilde{\mathbf{K}}_{i-1}. & (34) \end{aligned}$$

Substitution of these variables into the Kalman filter equations leads to

$$\begin{aligned} d\tilde{\mathbf{R}}_{e,i} &= \tilde{C}_i d\tilde{\mathbf{P}}_i \tilde{C}_i^T, & (35)\\ d\tilde{\mathbf{K}}_i &= \tilde{A}_i d\tilde{\mathbf{P}}_i \tilde{C}_i^T, & (36)\\ d\tilde{\mathbf{P}}_{i+1} &= \tilde{A}_i d\tilde{\mathbf{P}}_i \tilde{A}_i^T + \mathbf{\Psi}\tilde{\mathbf{K}}_{i-1}\tilde{\mathbf{R}}_{e,i-1}^{-1}\tilde{\mathbf{K}}_{i-1}^T\mathbf{\Psi}^T \\ &\quad -\tilde{\mathbf{K}}_i\tilde{\mathbf{R}}_{e,i}^{-1}\tilde{\mathbf{K}}_i^T. & (37) \end{aligned}$$

Assume that $d\tilde{\mathbf{P}}(i)$ can be factorized as follows (see Sayed *et al.* [10] );

$$d\tilde{\mathbf{P}}_i = \tilde{\mathbf{L}}_{i-1}\mathbf{M}_{i-1}\tilde{\mathbf{L}}_{i-1}^T, \tag{38}$$

with $\tilde{\mathbf{L}}_{i-1} \in \mathbb{R}^{n_m+n_s\times\alpha}$, $\mathbf{M}_{i-1} \in \mathbb{R}^{\alpha\times\alpha}$, $\alpha \ll n_m + n_s$, in which $\alpha$ the rank of matrix $\mathbf{M}_{i-1}$. When this factorization is used the matrices $\tilde{\mathbf{R}}_{e,i-1}$, $\tilde{\mathbf{K}}_{i-1}$ and $\tilde{\mathbf{L}}_{i-1}$ can be updated with a fast array algorithm. The matrix needed to accomplish the required transformations can be achieved by both a series of hyperbolic Givens rotations or hyperbolic Householder transformations, see [9].

The difference of the Riccati equation must have an as low as possible rank to make the algorithm significantly faster than the covariance form of the Kalman algorithm. By choosing the initial conditions properly, the rank for the Riccati difference equation can be reduced to $\alpha = 2N_uN_x$.

## 3 Output normal form parameterization

The state space description of the secondary path is rewritten to an output normal form parameterization. This parameterization has a few advantages in comparison to the full state space model. As shown in Ref. [11], not only the calculations needed to do the multiplications with the state matrices reduce due to the Hessenberg form of the state matrix, but the parameterization also makes it possible to solve the multiplications in a recursive way, leading to even more reduction of the floating point operations needed. Another benefit of the state space parameterization is the reduction of redundancy in the state matrices.

### 3.1 Transformation to output normal form

When the assumption that the secondary path doesn't change in time is met, the state matrices can be identified off-line. After identification the states can be transformed if the states are observable. An output norm form transforms the state matrices in such a way the observability Gramian is the identity matrix:

$$A_s^T A_s + C_s^T C_s = I. \tag{39}$$

When this is true, the states are orthogonal, giving several numerical advantages in comparison to the full state space model, in which the most important are the low round off noise gain and the notion that the amplitude of the signal is not changed throughout the filter (Roberts *et al.* [12]). The transformation of the state space model is done with a similarity transform matrix $T_t$. This matrix can be determined by calculating the solution $Q$ from the observability Gramian of the full state space system $A_s^T Q A_s + C_s^T C_s = Q$, by decomposing this solution ($Q = T_q T_q^T$) and calculating $T_t = T_q^{-T}$. The state matrices in output normal form can be calculated with $A_T = T_t A T_t^{-1}$, $B_T = T_t B$, $C_T = C T_t^{-1}$ and $D_T = D$. When these transformations are done, the columns of the matrix $\begin{bmatrix} C_T \\ A_T \end{bmatrix}$ are orthogonal. A second similarity transformation is done to transform the matrix to a Hessenberg form (the needed transformation matrix can be calculated with Given's rotations or Householder transformations). The resulting matrix $\begin{bmatrix} C_H \\ A_H \end{bmatrix}$ now can be decomposed with the following parameterization:

$$\begin{bmatrix} C_H \\ A_H \end{bmatrix} = T_1(\beta(1))....T(\beta(n)) \begin{bmatrix} 0 \\ I_n \end{bmatrix}. \tag{40}$$

In this equation $\beta_{AC} = \begin{bmatrix} \beta(1) & \dots & \beta(n) \end{bmatrix}^T$ is a vector with parameters ranging from $-1$ to 1. and $T_1.....T_n$ are rotation matrices, see [11].

### 3.2 Incorporation into the Kalman filter

To incorporate the output normal form parameterization into the Kalman filter in fast array form, a few adjustments have to be done, since the parameterization is only useful for the parts of the augmented state matrices that represent the secondary path. The matrix $\tilde{\mathbf{L}}_{i-1}$ has to be partitioned in the following way:

$$\tilde{\mathbf{L}}_{i-1} = \begin{bmatrix} \tilde{L}_{p_1,i-1} \\ \tilde{L}_{p_2,i-1} \end{bmatrix}, \tag{41}$$

in which $\tilde{L}_{p_1,i-1} \in \mathbb{R}^{n_m \times 2N_u N_x}$, and $\tilde{L}_{p_2,i-1} \in \mathbb{R}^{n_s \times 2N_u N_x}$. The parameterization can now be used to calculate

$$\begin{bmatrix} C_H \\ A_H \end{bmatrix} \tilde{L}_{p_2,i-1} = T_1(\beta(1))....T(\beta(n)) \begin{bmatrix} 0 \\ \tilde{L}_{p_2,i-1} \end{bmatrix}. \tag{42}$$

Since the calculation of $C_i \tilde{\mathbf{L}}_{i-1}$ also requires $\tilde{L}_{p_1,i-1}$, the end result is calculated with

$$C_i \tilde{L}_{i-1} = D_s(I_{N_u} \otimes \mathbf{x}_{n_w+1}^T(i)) \tilde{L}_{p_1,i-1} + C_H \tilde{L}_{p_2,i-1.} \tag{43}$$

Also the $A$ matrix has to be partitioned into $A_i = \begin{bmatrix} A_{p_1,i} \\ A_{p_2,i} \end{bmatrix}$ with $A_{p_1,i} \in \mathbb{R}^{n_m \times n_m + n_s}$ and $A_{p_2,i} \in \mathbb{R}^{n_s \times n_m + n_s}$ . The matrix multiplication $A_i \tilde{\mathbf{L}}_{i-1}$ can now be executed with

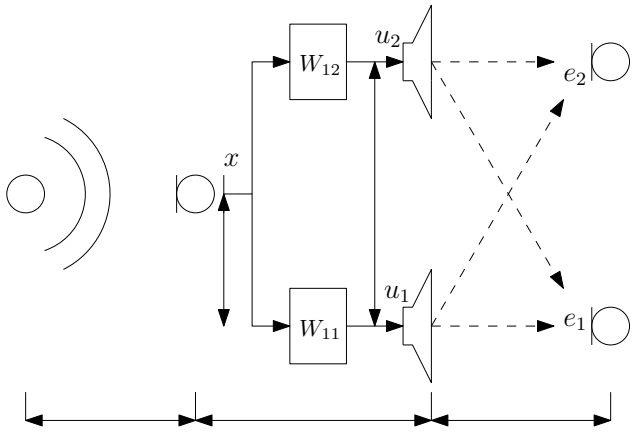


Figure 3: MIMO simulation setup. The horizontal distances indicated by the arrows are, from left to right: 5 m, 5m, 5m. The vertical distance between the controlled sources is 5m. The reference sensor and noise source are positioned symmetrically between the two controlled sources.

$$A_{p_1,i}\tilde{\mathbf{L}}_{i-1} = \left[\begin{array}{cc} \lambda^{-1/2} I_{n_m} & 0_{n_m \times n_s} \end{array}\right] \tilde{\mathbf{L}}_{i-1}. \tag{44}$$

$$A_{p_2,i}\tilde{\mathbf{L}}_{i-1} = B_s(I_{N_u} \otimes \mathbf{x}_{n_w}^T(i))\tilde{L}_{p_1,i-1} + A_H \tilde{L}_{p_2,i-1}, \tag{45}$$

Also the update equations can be rewritten to a more efficient form by using a dummy update vector.

$$\left[\begin{array}{c} \hat{e}_d \\ \hat{x}_d \end{array}\right] = \left[\begin{array}{c} C_H \\ A_H \end{array}\right] \hat{\boldsymbol{\theta}}_i = T_1(\beta(1))....T(\beta(n)) \left[\begin{array}{c} 0 \\ \hat{\boldsymbol{\theta}}_i \end{array}\right], \tag{46}$$

$$\boldsymbol{\epsilon}_i = e_i - \hat{e}_d \in \mathbb{R}^{N_e}, \tag{47}$$

$$\hat{\boldsymbol{\theta}}_{i+1} = \hat{x}_d + \overline{K}_i^{\theta} R_{e,i}^{-1/2} \boldsymbol{\epsilon}_i \in \mathbb{R}^{n_s}. \tag{48}$$

When these alternate equations are used, only the parameters $\beta_{AC} = \left[\begin{array}{ccc} \beta(1) & \ldots & \beta(n) \end{array}\right]^T$ are needed, instead of the state matrices $A_s$ and $C_s$.

## 4 Simulation results

A simple MIMO system was simulated to check the behavior of the filter. The simulation setup is shown in Fig. 3. The acoustic pressure generated by the noise source on the left hand side is to be minimized at the error microphones with error signals $e_1$ and $e_2$ by choosing appropiate filter coefficients for $W_{11}$ and $W_{12}$ which drive the controlled sources. The system is assumed to be in free space, without reflecting boundaries. The controlled sources are point sources having constant volume velocity. There is no feedback from the controlled sources to the reference signal $x(i)$. The transformation matrix is calculated with a series of hyperbolic Housholder transformations at every instance $i$. The noise source is emitting white noise with a cut-off frequency at 1000 Hz and the system is sampled with a frequency of $f_s = 2000$ Hz.

Due to symmetry, it is expected that the convergence curves of $e_1$ and $e_2$ will be the same. As can be seen in Fig. 4, this is the case (the filter is turned on after 1000 samples).

Additional numerical tests were carried out with recorded signals from accelerating vehicles which showed that the Kalman filter copes well with the changing spectrum with no noticeable change in the residual signal as function of time after convergence. Tracking behavior of the Kalman filter with moving sound sources including the Doppler effect is described in Ref. [8].

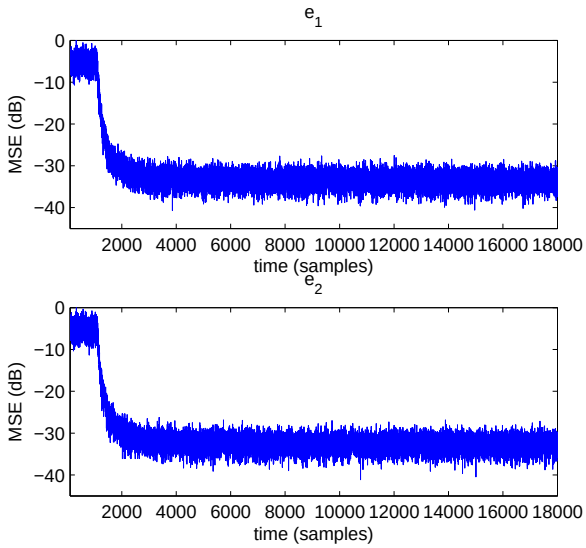


Figure 4: Convergence curve of a symmetric MIMO system. The filter is turned on after 1000 samples.

Figure 5: Experimental setup consisting of a duct for active noise control tests.

## 5 Experimental results

A duct was used for testing the Kalman filter in an Active Noise Control environment. A picture of the setup is shown in Fig. 5. A loudspeaker on the left hand side generates a noise source and the goal is the control the signal from the secondary source near the other end of the duct in such a way that the signal on the error microphone measures is reduced. The error sensor is placed at at the other end of the tube. The control system was implemented on an embedded PC, specifically made for ANC purposes, running an Linux operating system. A detailed description of the system can be found in [13]. The noise signal was created on the embedded PC and was also used as reference signal, so no feedback compensation was needed. System identification of the secondary path was done beforehand to determine the proper initial conditions of the filter with subspace identification. For a system with a sampling frequency of $f_s$ =2000 Hz with a 30th order plant, a Variance Accounted For (VAF) value of around 99.8% could be achieved. The regularization coefficient $\delta$ was chosen in such a way that no overshoot before convergence was present, but the convergence rate was as fast as possible. The estimated covariance of the noise terms were set at approximately 20 dB below the variance of the nominal reference signal. The forgetting factor was set to $\lambda = 1$. The number of filter coefficients was set to varying values, depending on the structure of the state space model.

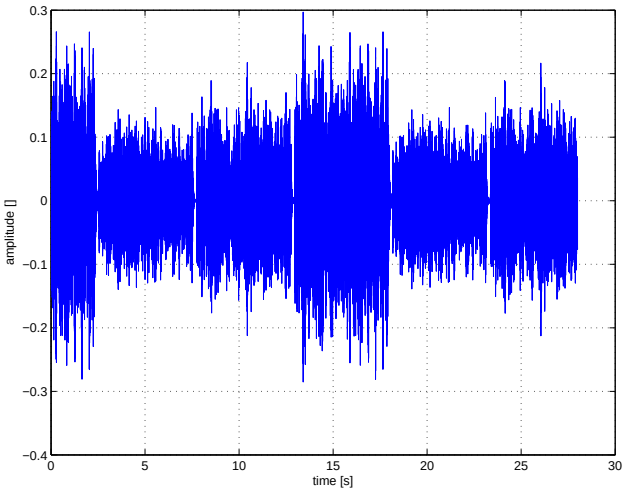


Figure 6: Disturbance signal.

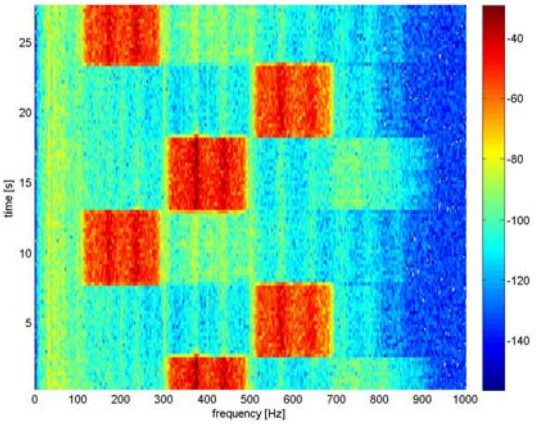


Figure 7: Spectrogram of the disturbance signal; colorscale in dB.

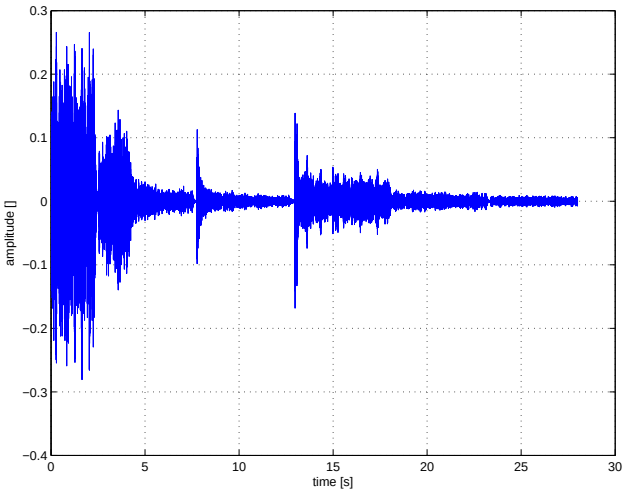


Figure 8: Error signal.

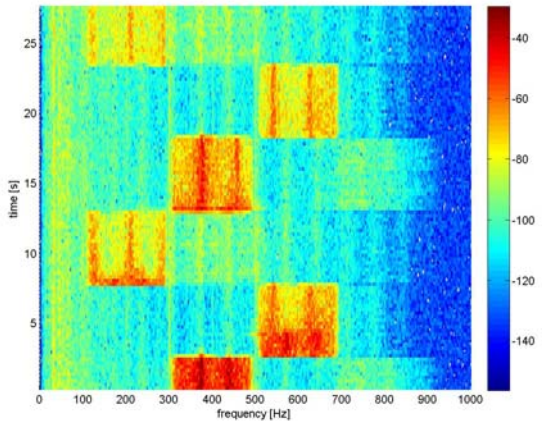


Figure 9: Spectrogram of the error signal; colorscale in dB.

## 5.1 Influence of secondary path models

Use of the full state space model of the secondary path was unsuccesful. Firstly, at the 2 kHz sample rate the Kalman filter exceeded the maximum available processing resources of the platform. At a lower sample rate of 500 Hz the processing resources were sufficient but still the algorithm failed to converge, constituting a more fundamental problem. It was found that the model of the secondary path prevented a proper operation of the algorithm. Therefore two alternatives were evaluated, a state space model representing an FIR filter structure and a state space model in output normal form, as described in Section 3. When an FIR structure is used for a state space model, the Kalman algorithm can be implemented efficiently for a SISO system, since a large part of the estimated state matrices will consist of zeros. Therefore, an FIR length of 300 coefficients could be achieved without a problem. In the duct, a maximum broadband reduction of about 12 dB could be achieved. The rate of convergence was comparable to the results achieved in the simulations. The filter also proved to be quite robust to rounding-off errors. To test the robustness of the algorithm, the noise control system was turned on and after four hours it was checked if the system was still working correctly. This was found to be the case.

## 5.2 Output normal form structure

The output normal form was found to require less computational effort and to be numerically more robust than the full state space system. Although the description of the state space model in output normal form may not be as efficient as the FIR filter structure for SISO systems, it is expected to be more efficient for MIMO systems, since common dynamics in the filter path will only be present in the output normal form state space model once, after the redundancy will be removed, in contrast to a MIMO FIR state space model. Also the noise reduction at the error microphone is better, being approximately 14 dB, since the accuracy of the secondary path models of the output normal parameterization can be higher for a given computational complexity.

## 5.3 Band-limited noise

To test the perfomance of the filter for changing noise spectra, three non-overlapping bandpass filters where chosen to filter the white noise and every two seconds the bandpass filter was switched. The scalar disturbance signal $\mathbf{d}_i$ is depicted in Fig. 6 and the corresponding spectrogram in Fig. 7. The resonances in the duct can be identified in the spectrogram. The scalar error signal $\mathbf{e}_i$ with control system switched on can be found in Fig. 8 and the corresponding spectrogram in Fig. 9. The first time the filter is confronted with

new information (the first three bands), the filter needs a small amount of time to converge at every switch. When the noise bands return, the filter doesn't need any time at all to converge. The same behavior can be seen in simulations. Since the geometry of the duct causes resonances at certain frequencies, the FIR controller adapts its filter coefficients in such away that these resonances are included into the model. When new frequency information is present in the signal, it needs to tune the FIR coefficients, so that the resonances at these frequencies are included, but the information at other frequencies is still present in the model, due to the recursive nature of the filter. When the frequency response of the FIR filter is observed as function of time, it can be seen that the filter not only adds information for the new frequencies, every time a new frequency band occurs, but also keeps the old information available.

## Acknowledgements

The authors would like to thank Geert Jan Laanstra and Henny Kuipers of University of Twente, Signals and Systems group, Faculty EEMCS for the support and Andre de Boer for hosting the Master's assignment in his group.

## References

[1] S.J. Elliott. *Signal Processing for Active Control*. Academic Press, 2001.

[2] E. Bjarnason. Active noise cancellation using a modified form of the filtered-x lms algorithm. *Proc. of Eusipco 92, 6th Europ. Sign. Proc. Conf.*, pages 1053–1056, 1992.

[3] S.J. Flockton. Fast adaption algorithms in active noise control. *Sec. Conf. on Rec. Advanc. in Act. Noise Contr. of Sound and Vibr.*, pages 802–810, Apr. 1993.

[4] A.H. Sayed and T. Kailath. A state space approach to adaptive rls filtering. *IEEE Sign. Process. Mag.*, pages 18–60, 1994.

[5] R. Fraanje. *Robust and Fast Schemes in Broadband Active Noise and Vibration Control*. PhD thesis, University of Twente, 2004.

[6] K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, Upper Saddle River, New Jersey 07458, 1996.

[7] B. Sayyarrodsari, J.P. How, B. Hassabi, and Alain Carrier. Estimation-based synthesis of $h_\infty$-optimal adaptive fir filters for filtered-lms problems. *IEEE Trans. on Sign. Process.*, 49.NO.1:164–178, 2001.

[8] S. van Ophem and A.P. Berkhoff. Performance of a multi-channel adaptive kalman algorithm for active noise control of non-stationary sources. In *Proc. Internoise 2012*. INCE, 2012.

[9] A.H. Sayed. *Fundamentals of Adaptive Filtering*. John Wiley & Sons Inc., 2003.

[10] A.H. Sayed and T. Kailath. Extended chandrasekhar recursions. *IEEE Trans. on Autom. Contr.*, 39, NO. 3:619–623, 1994.

[11] M. Verhaegen and V. Verdult. *Filtering and System Identification: A least Squares Approach*. Cambridge University Press, 2007.

[12] R. A. Roberts and C. T. Mullis. *Digital Signal Processing*. Addison-Wesley, 1987.

[13] A.P. Berkhoff and J.M. Wesselink. Combined mimo adaptive and decentralized controllers for broadband active noise and vibration control. *Mechanical Systems and Signal Processing*, 25:1702–1714, 2011.